MARCH MACHINE LEARNNG MANIA 2021-NCAAW SPREAD

In stage one of this two-stage competition, participants will build and test their models against previous tournaments. In the second stage, participants will predict the point spreads of the 2021 tournament. We don't need to participate in the first stage to enter the second. The first stage exists to incentivize model building and provide a means to score predictions. The real competition is forecasting the 2021 results.

To Predict

Stage 1 - You should submit predicted point spread for every possible matchup in the past 5 NCAAW® tournaments (seasons 2015-2019).

Stage 2 - You should submit predicted point spread for every possible matchup before the 2021 tournament begins.

Dataset

WNCAATourneyDetailedResults.csv
This file provides team-level box scores for many NCAA® tournaments, starting with the 2010 season. All games listed in the WNCAATourneyCompactResults file since the 2010 season should exactly be present in the WNCAATourneyDetailedResults file.

submit_stage1_elo_tuned.csv

elo_2021-04-05.csv

elo_calibrated_margin_submitted.csv

2021NCAAWTourneyMarginResults.csv

elo_calibrated_margin_fix_season_merge.csv

Step 1: Importing the libraries

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import matplotlib.pyplot as plt

from pathlib import Path

from sklearn.neighbors import KNeighborsRegressor
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import GridSearchCV
```

Regression based on k-nearest neighbors.The target is predicted by local interpolation of the targets associated of the nearest neighbors in the training set.

Mean squared error regression loss.

GridSearchCV implements a “fit” and a “score” method. It also implements “predict”, “predict_proba”, “decision_function”, “transform” and “inverse_transform” if they are implemented in the estimator used.

The parameters of the estimator used to apply these methods are optimized by cross-validated grid-search over a parameter grid.

Step 2: Loading the data

```
1 from google.colab import files
2 uploaded=files.upload()
```

Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

```
Saving WNCAATourneyDetailedResults.csv to WNCAATourneyDetailedResults.csv
```

```
1 from google.colab import files
2 uploaded=files.upload()
```

Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

```
Saving submit_stage1_elo_tuned.csv to submit_stage1_elo_tuned.csv
```

Step 3: add T1/2_TeamID to the tournament results.

```
for ii, row in tourney_results.iterrows():
    t1_id, t2_id = row.WTeamID, row.LTeamID
    if t1_id > t2_id:
        t2_id, t1_id = t1_id, t2_id
    tourney_results.loc[ii, "T1_TeamID"] = t1_id
    tourney_results.loc[ii, "T2_TeamID"] = t2_id
    tourney_results.loc[ii, "Margin"] = row.Margin
tourney_results["T1_TeamID"] = tourney_results["T1_TeamID"].astype(int)
tourney_results["T2_TeamID"] = tourney_results["T2_TeamID"].astype(int)
tourney_results.head(10)
```

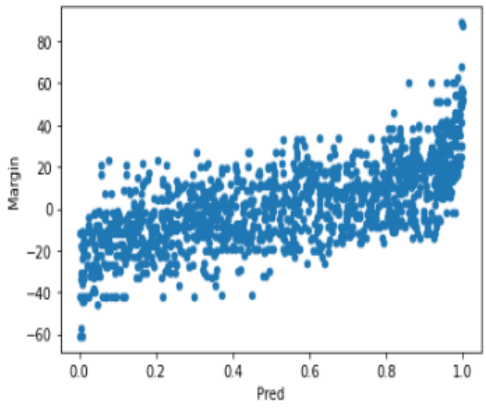
	Season	DayNum	WTeamID	WScore	LTeamID	LScore	WLoc	NumOT	WFGM	WFGA	...	LOR	LDR	LAst	LTO	LStl	LBlk	LPF	Margin	T1_TeamID	T2
0	2010	138	3124	69	3201	55	N	0	28	57	...	17	19	12	18	4	1	18	14	3124	
1	2010	138	3173	67	3395	66	N	0	23	59	...	18	26	8	8	8	6	22	1	3173	
2	2010	138	3181	72	3214	37	H	0	26	57	...	10	21	4	16	6	4	20	35	3181	
3	2010	138	3199	75	3256	61	H	0	25	63	...	16	21	13	16	5	4	24	14	3199	
4	2010	138	3207	62	3265	42	N	0	24	68	...	16	22	9	10	3	4	12	20	3207	
5	2010	138	3208	64	3408	59	N	0	21	50	...	17	17	11	12	11	3	20	5	3208	
6	2010	138	3211	82	3314	76	H	0	36	75	...	17	31	15	18	12	5	16	6	3211	
7	2010	138	3234	70	3353	63	N	0	19	44	...	11	18	13	13	10	2	21	7	3234	
8	2010	138	3246	83	3251	77	H	0	25	49	...	17	21	8	22	8	1	25	6	3246	
9	2010	138	3261	60	3216	39	N	0	18	56	...	11	25	12	17	7	5	20	-21	3216	

10 rows × 37 columns

Step 4: stage1 submission with the tournament results.

```
merged = prob_df.merge(tourney_results, on=["T1_TeamID", "T2_TeamID"])
merged.plot("Pred", "Margin", kind="scatter");
```

```
/usr/local/lib/python3.9/dist-packages/pandas/plotting/_matplotlib/core.py:1114: UserWarning: No data for colormapping provided
via 'c'. Parameters 'cmap' will be ignored
  scatter = ax.scatter(
```



Step 5: Checking the Final Score

```
from google.colab import files
uploaded=files.upload()
```

Choose Files No file chosen

Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

```
Saving elo_calibrated_margin_submitted.csv to elo_calibrated_margin_submitted.csv
```

```
stage2_df = pd.read_csv("elo_2021-04-05.csv")
pred_margin = clf.best_estimator_.predict(stage2_df.Pred.to_numpy().reshape(-1, 1))
stage2_df["Pred"] = pred_margin
stage2_df[["ID", "Pred"]].to_csv("elo_calibrated_margin_submitted.csv", index=False)
```

```
from google.colab import files
uploaded=files.upload()
```

Choose Files No file chosen

Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

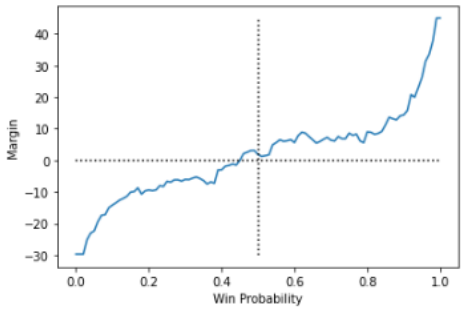
```
Saving 2021NCAAWTourneyMarginResults.csv to 2021NCAAWTourneyMarginResults.csv
```

```
stage2_result = pd.read_csv("2021NCAAWTourneyMarginResults.csv")
pred_results_combined = stage2_result.merge(stage2_df, on="ID", how="left")
print(f"Final score: {mean_squared_error(pred_results_combined.Margin, pred_results_combined.Pred, squared=False)}")
```

```
Final score: 11.273641659312501
```

Step 6: Plotting the Model

```
x = np.linspace(0, 1, 101)
pred = clf.best_estimator_.predict(x.reshape(-1, 1))
plt.plot(x, pred)
plt.xlabel("Win Probability")
plt.ylabel("Margin")
plt.hlines(0, 0, 1, linestyle="dotted", colors="k")
plt.vlines(0.5, -30, 45, linestyle="dotted", colors="k");
```



There is a lot of noise in these competitions and to some degree I got l ucky with the asymmetry. Although there is just a chance asymmetry in the data as well. Alphabetically lower teams in the NCAAW are jus t a little stronger than alphabetically higher teams.